Amendments to the Claims:

This listing of claims replaces all prior versions and listings of claims in the application:

Listing of Claims:

1. (Currently Amended) A computer-implemented method for identifying one or more objects within an image, the method comprising:

receiving an image that includes two or more non-overlapping embedded images~~at least one object~~;

identifying a plurality of edge pixels in the image~~, an edge pixel~~ based on a respective gradient value associated with each of the plurality of edge pixels; ~~being a pixel that borders two contrasting areas of the image, the plurality of edge pixels including both inner edge pixels and outer edge pixels;~~

selecting an edge pixel from the plurality of edge pixels;

identifying a plurality of substantially connected edge pixels being substantially connected to the selected edge pixel; and

identifying a bounding area within the image, the bounding area surrounding ~~finding a~~ the plurality of substantially connected edge pixels.~~connected component correlated with each object, the substantially connected component comprising a set of the edge pixels that are connected by traversing substantially only edge pixels.~~

2. (Cancelled)

3. (Currently Amended) The method of claim 1, wherein identifying a plurality of edge pixels includes computing [[a]] the respective gradient value for each of a plurality of pixels in the image.

4. (Currently Amended) The method of claim [[2]] 3, wherein computing the respective gradient value for each of the ~~a given~~ plurality of pixels includes, for each pixel, comparing respective pixel colors ~~in a~~ of a neighborhood of pixels surrounding ~~the given~~ each pixel.

5. (Currently Amended) The method of claim [[2]] 3, wherein computing the respective gradient value for each of the ~~a given~~ plurality of pixels includes using an image smoothing filter to filter noise from the image.

6. (Currently Amended) The method of claim 1, further comprising passing each ~~component~~ plurality of substantially connected edge pixels to a processor that extracts ~~the~~ a location of an embedded image from the two or more non-overlapping images ~~the object from the component~~.

7. (Currently Amended) The method of claim [[5]] 6, further comprising refining the extracted location.

8. (Currently Amended) The method of claim [[6]] 7, further comprising using the extracted location to crop the embedded image ~~object~~ from the image.

9. (Currently Amended) The method of claim 1, further comprising splitting the bounding area of the image into a first of the two or more non-overlapping embedded images and a second of the two or more non-overlapping embedded images.~~a component into two components.~~

10. (Currently Amended) The method of claim 1, further comprising merging the bounding area within the image with another bounding area within the image~~two components~~ into a single ~~component~~bounding area.

11. (Currently Amended) The method of claim 1, further comprising:

extracting ~~the~~ a location of each ~~object~~ of the two or more non-overlapping embedded images from the image; and

using the ~~extracted object~~ location to seed a crop operation.

12. (Currently Amended) The method of claim [[10]]11, wherein using the ~~extracted object~~ location to seed a crop operation includes:

for each ~~object~~ of the two or more non-overlapping embedded images in the image, using the ~~extracted object~~ location to define a cropping area; and

cropping all the defined cropping areas in a single cropping operation.

13. (Currently Amended) The method of claim [[11]] 12, wherein:

the ~~extracted object~~ location specifies ~~how~~ an alignment of one of the two or more non-overlapping embedded images ~~the object is aligned~~ with respect to the image; and

using the ~~extracted object~~ location to define a cropping area includes using the alignment of one of the two or more non overlapping embedded images~~of the object~~ to define ~~the~~ an alignment of the cropping area.

14. (Currently Amended) The method of claim [[11]] 12, further comprising:

prior to cropping all the defined cropping areas, adjusting one or more of the defined cropping areas in response to user input.

15. (Currently Amended) The method of claim [[13]] 12, further comprising:

prior to cropping all the defined cropping areas, ~~wherein adjusting one or more of the defined cropping areas includes~~ merging two of the defined cropping areas into a single defined cropping area in response to user input.

16. (Currently Amended) The method of claim [[13]] 14, wherein adjusting one or more of the defined cropping areas includes splitting a single cropping area into two or more cropping areas.

17. (Currently Amended) A computer program product, tangibly stored on a computer-readable medium, for identifying one or more objects within an image, comprising instructions operable to cause a programmable processor to perform operations comprising:

receiving an image that includes two or more non-overlapping embedded images~~at least one object~~;

identifying a plurality of edge pixels in the image~~, an edge pixel~~ based on a respective gradient value associated with each of the plurality of edge pixels; ~~being a pixel that borders two contrasting areas of the image, the plurality of edge pixels including both inner edge pixels and outer edge pixels;~~

selecting an edge pixel from the plurality of edge pixels;

identifying a plurality of substantially connected edge pixels being substantially connected to the selected edge pixel; and

identifying a bounding area within the image, the bounding area surrounding ~~finding a~~ the plurality of substantially connected edge pixels.~~connected component correlated with each object, the substantially connected component comprising a set of the edge pixels that are connected by traversing substantially only edge pixels.~~

18. (Cancelled)

19. (Currently Amended) The product of claim [[16]] 17, wherein identifying a plurality of edge pixels includes computing [[a]] the respective gradient value for each of a plurality of pixels in the image.

20. (Currently Amended) The product of claim [[18]] 19, wherein computing the respective gradient value for each of the ~~a given~~ plurality of pixels includes, for each pixel, comparing respective pixel colors ~~in a~~ of a neighborhood of pixels surrounding ~~the given~~ each pixel.

21. (Currently Amended) The product of claim [[18]] 19, wherein computing the respective gradient value for each of the ~~a given~~ plurality of pixels includes using an image smoothing filter to filter noise from the image.

22. (Currently Amended) The product of claim [[16]] 17, wherein the operations further comprise passing each ~~component~~ plurality of substantially connected edge pixels to a processor that extracts ~~the~~ a location of an embedded image from the two or more non-overlapping images ~~the object from the component~~.

23. (Currently Amended) The product of claim [[21]] 22, wherein the operations further comprise refining the extracted location.

24. (Currently Amended) The product of claim [[22]] 23, wherein the operations further comprise using the extracted location to crop the ~~object~~ embedded image from the image.

25. (Currently Amended) The product of claim [[16]] 17, wherein the operations further comprise splitting the bounding area of the image into a first of the two or more non-overlapping embedded images and a second of the two or more non-overlapping embedded images.~~a component into two components.~~.

26. (Currently Amended) The product of claim [[16]] 17, wherein the operations further comprise merging the bounding area within the image with another bounding area within the image~~two components~~ into a single ~~component~~bounding area.

27. (Currently Amended) The product of claim [[16]] 17, wherein the operations further comprise:

extracting ~~the~~ a location of each ~~object~~ of the two or more non-overlapping embedded images from the image; and

using the ~~extracted object~~ location to seed a crop operation.

28. (Currently Amended) The product of claim [[26]] 27, wherein using the ~~extracted object~~ location to seed a crop operation includes:

for each ~~object~~ of the two or more non-overlapping embedded images in the image, using the ~~extracted object~~ location to define a cropping area; and

cropping all the defined cropping areas in a single cropping operation.

29. (Currently Amended) The product of claim [[27]] 28, wherein:

the ~~extracted object~~ location specifies ~~how~~ an alignment of one of the two or more non-overlapping embedded images ~~the object is aligned~~ with respect to the image; and

using the ~~extracted object~~ location to define a cropping area includes using the alignment of one of the two or more non overlapping embedded images~~of the object~~ to define ~~the~~ an alignment of the cropping area.

30. (Currently Amended) The product of claim [[27]] 28, wherein the operations further comprise:

prior to cropping all the defined cropping areas, adjusting one or more of the defined cropping areas in response to user input.

31. (Currently Amended) The product of claim [[29]] 28, wherein the operations further comprise:

prior to cropping all the defined cropping areas, ~~wherein adjusting one or more of the defined cropping areas includes~~ merging two of the defined cropping areas into a single defined cropping area in response to user input.

32. (Currently Amended) The product of claim [[29]] 30, wherein adjusting one or more of the defined cropping areas includes splitting a single cropping area into two or more cropping areas.

33. (Currently Amcnded) A computer program product, tangibly stored on a computer-readable medium, for identifying multiple objects within a scanned image, comprising instructions operable to cause a programmable processor to perform operations comprising:

receiving a scanned image that includes multiple non-overlapping embedded images~~objects~~; and

identifying the multiple non-overlapping embedded images ~~objects~~ by:

(1) generating an edge pixel map of the image based on a respective gradient value associated with each pixel in the scanned image, the edge pixel map identifying each pixel in the scanned image as being an edge pixel or a non-edge pixel, an edge pixel being a pixel ~~that borders two contrasting areas of the image~~having a gradient value above a threshold value;

(2) scanning the edge pixel map until an edge pixel is found;

(3) ~~computing~~ identifying a connected component containing the edge pixel, the connected component being a set of edge pixels that are connected by traversing only adjacent edge pixels;

(4) extracting one of the multiple non-overlapping embedded images~~objects from the~~ based on the connected component;

(5) erasing from the edge pixel map all the ~~edge~~ pixels that belong to the connected component or that correspond to pixels inside ~~are enclosed by~~ the extracted one of the multiple non-overlapping embedded images~~object~~; and

(6) repeating steps (2) through (5) until no more edge pixels are found.